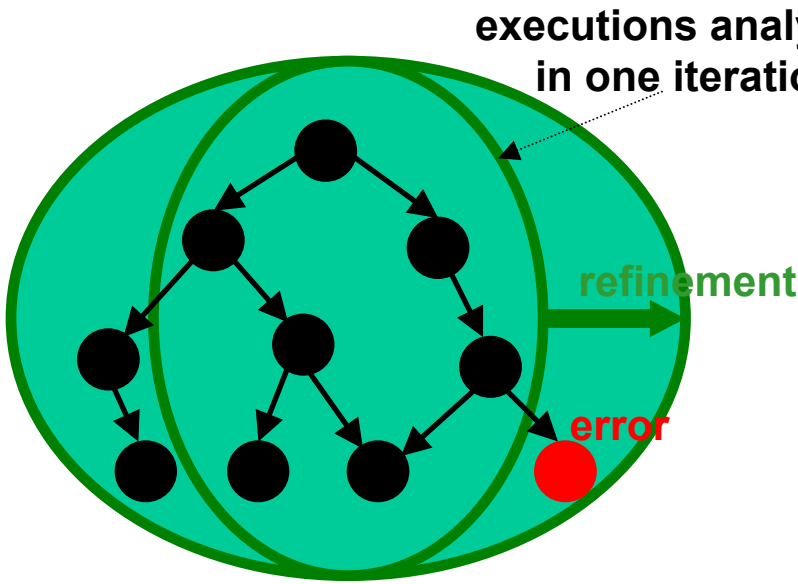
Efficient Software Verification Method Developed at Ames

Verification is essential for ensuring highly reliable software for NASA missions. Testing alone is not enough to uncover subtle interaction errors and to produce traces that exhibit those errors.

We developed a new software verification method for efficient error detection.

- The method is **automatic** and reports counterexamples exhibiting errors (uses model checking techniques)
- It can **scale** to the analysis of large systems (uses abstraction techniques to perform approximate analysis)
- It is **precise**: it does not report false positives (it analyzes only feasible system executions)

System under analysis



Innovation

The novelty of the method is the integration of an abstract analysis with concrete program execution, and the use of a theorem prover to detect (possible) incompleteness of abstract analysis and to iteratively refine the analysis.

The technique **finds errors faster** and it is **more efficient** (in the number of theorem prover calls) than the standard (over-approximation) abstract model checking methods.

Efficient Software Verification Method Developed at Ames

- **POC:** Corina Pasareanu (QSS, RSE, Code TI, pcorina@email.arc.nasa.gov), Willem Visser (RIACS/USRA, RSE, Code TI, wvisser@email.arc.nasa.gov)
- **Collaborator:** Radek Pelanek (Masayk University, Brno, Czech Republic)

- **Funding:** ITSR Legacy Funding, QSS Summer Intern Program

- **Background:** One of the main goals of RSE's research is to develop automated, robust tools and techniques for efficient analysis of mission software.
- **Accomplishment:** We have developed an iterative error detection method that analyzes efficiently and systematically increasing portions of the executions of a software system. The method relies on abstraction and theorem prover techniques to detect automatically incompleteness of the analysis and to refine the analysis (so that at the next iteration, more system behaviors are analyzed, thus increasing the chance of detecting errors). In comparison with previous software verification approaches, which use (over-approximation) abstraction and model checking, our method **finds errors faster and it is more efficient** (in the number of theorem prover calls). The paper describing the method and its applications was presented at the 17th International Conference on Computer Aided Verification (CAV), Edinburgh, UK, July 2005 (paper title: "Concrete Model Checking with Abstract Matching and Refinement").
- **Benefits:** The method has been applied successfully for error detection in RAX (Remote Agent Experiment) – a component extracted from a NASA embedded spacecraft control application, which deadlocked in space (on the Deep Space 1 spacecraft). In the future, we plan to apply the method to the analysis of other NASA mission software (e.g. the next generation of robust execution systems).

The method represents a new pragmatic approach to software verification and it is currently under consideration for a NASA Invention Disclosure (NF-1679 submitted).